

# Satogaeri

*Markus Leitner*

## Contents

1	Introduction . . . . .	1
2	Satogaeri . . . . .	1
2.1	Rules . . . . .	1
3	Satisfiability Modulo Theories . . . . .	2
3.1	Lazy SMT . . . . .	3
3.2	SMT-Library . . . . .	3
3.3	Logics . . . . .	3
3.4	CVC4 . . . . .	3
4	The Solver . . . . .	3
5	The Generator . . . . .	3
5.1	Drawing . . . . .	3
6	Statistics . . . . .	3
7	JavaFX . . . . .	3
8	Conclusion . . . . .	3

## 1 Introduction

## 2 Satogaeri

Satogaeri is a logic puzzle published by the Japanese company Nikoli Co.m Ltd. in 2002. Satogaeri only made a booklet appearance in Nikoli Vol. 99, Vol. 100 and Vol. 101. However it got revived on nikoli.com in 2013.

### 2.1 Rules

The official rules from Nikoi state:

1. The areas enclosed by bold lines, are called "Countries." Move the circles, vertically or horizontally, so each country contains only one circle.
2. The numbers in the circles indicate how many cells they have to pass through. Circles without numbers may move any distance, but some of them do not move.
3. The circles cannot cross the tracks of other circles and cannot go over other circles.

And additionally the 4th rule, which counts for every logic puzzle:

- 4 A puzzle only has one solution.

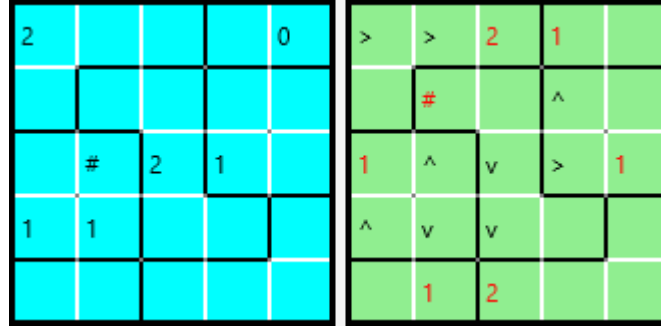


Fig. 1: Small sample puzzle with its solution

Handcrafted puzzles tend to have a mirrored country-layout either vertically, horizontally or both as can be seen in Figure 1. Numberless Circles are here referred to as '#’.

### 3 Satisfiability Modulo Theories

Before we can go on and talk about how to solve Satogaeri problems we first have to look into Satisfiability Modulo Theories (SMT) for this was used to create the solver and the generator.

SMT can be seen as an enhancement of the traditional Satisfiability (SAT) solving. In SAT solving one tries to find an interpretation of a given problem, where this interpretation is a boolean formula composed of boolean variables and expressions like AND, OR and NOT to state a few. If this constructed formula has a configuration of its variables where the formula evaluates to 'true', then also the original problem must have a solution. A satisfying configuration of the variables often gives a good clue on how to solve the original problem. To state an example: the formula

$$(a \vee b \vee (\neg c)) \iff ((\neg a) \wedge c)$$

is satisfied with the assignment  $a$  is 'false',  $b$  and  $c$  are 'true'. Therefore the originating problem of this formula must be solvable too.

**3.1 Lazy SMT****3.2 SMT-Library****3.3 Logics****3.4 CVC4****4 The Solver****5 The Generator****5.1 Drawing****6 Statistics****7 JavaFX****8 Conclusion**